

Unit-testing v .NET od A do Z

Robert Haken | [TechEd 2021](#)

Video

Unit-test

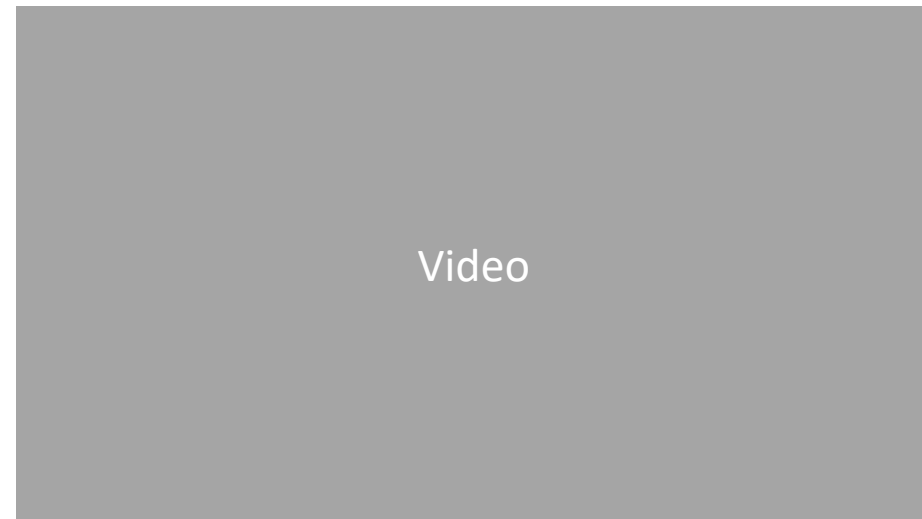
*Automatizovaný test
napsaný vývojářem k ujištění,
že určitý stavební dílek aplikace
splňuje design a funguje, jak bylo zamýšleno.*

[demo](#)

Video

DEMO

PriceResolverTests



Video

Proč unit-testing?

rychlejší feedback-loop pro detekci chyb

předcházení regresím

usnadnění vývoje

lepší strukturování kódu

Video

Jak na to?

Test frameworks

[MSTest - xUnit - NUnit](#)

State-based testing (result-based)

[demo](#) [demo](#)

Interaction-based testing

[demo](#) [demo](#)

Good unit-test

automated and **repeatable**

easy to implement + **maintainable**

relevant tomorrow

easy to run + runs quickly

consistent in its results

has **full control** of the unit under test

isolated

readable - easy to detect what was expected
and to locate the problem

Video

Best-practices

Test project per project (Xy.Tests)

+ IntegrationTests + TestsForLocalDebugging

Test class per tested class (SUT)

Team Culture = **all green**

Continuous Integration

Video

Best-practices

KISS

Arrange-Act-Assert / Given-When-Then

Small

Clear (Cyclomatic complexity: 1)

See it failing!!!

Video

Dependencies

Video

Dependencies

„stub“ = controllable replacement for dependency demo demo

„seams“ = places in code to plug in different functionality

- constructor parameter
- public settable property
- virtual method override
- delegate
- method parameter
- ambient context



Video

Mocking

„Stub“ = controllable replacement for dependency

„Mock“ = fake object with interaction verification

„Fakes“ = stubs + mocks

Moq

- Setup demo
- Callback demo
- Verify demo
- Callbase = true demo
- Strict / Loose

Video

Test smell

Video

Test smell

Logic in tests

Overspecifying tests

Testing multiple aspects

Testing internal state

Duplicate tests

Ordered tests

Video

Test smell 2

Tests sharing state

Long-running tests

Tests with variables

- `DateTime.Now`
- `Random()`

Unreadable tests

Video

Techniques

Video

Techniques & Tips

Parametrized tests [demo](#) [demo](#) [demo](#)

SUT Factory [demo](#)

Fixtures [demo](#)

Bug repro [demo](#)

Mocking time [demo](#)

Ambient context, Initialize, Cleanup [demo](#)

Exception details [demo](#)

Custom mock (e.g. HTTP) [demo](#)

Race conditions (poor-mans) [demo](#)

Video

Test-Driven-Development [TDD]

1. Write test
2. See it failing (RED)
3. Implement the system to pass the test (GREEN)
4. Refactor the system to a better shape (GREEN)

Video

Reference

DEMO - <https://github.com/hakenr/UnitTestingDemo>

Blog: <https://knowledge-base.havit.cz/>

Twitter: [@RobertHaken](https://twitter.com/RobertHaken)

Video

Tech Ed

A small white icon of a microchip or processor, positioned between the words 'Tech' and 'Ed' in the main title.

Děkuji Vám za pozornost

Přednášející je Vám k dispozici na chatu ještě 15 minut po skončení přednášky.